**NORTH SOUTH UNIVERSITY**

**Department of Electrical and Computer Engineering**



**CSE 115**

**Programming Language I**

**Section-02**

**Project Title: Periodic Table**

**Submitted to: Ashik Mostafa Alvi**

**Date of Submission: 26th April 2019**

**Name : Md. Saeem Hossain Shanto ID: 1912218642**

**Name : Nazmul Hasan ID: 1911742042**

**Project Overview :**

Chemistry is everywhere in the world around us! Chemistry sometimes is called the "central science" and to know chemistry, all we need to learn about the chemical elements which occur reactions behind. Periodic Table is one of the most important part of Chemistry. In this C language program where the user, can see a look alike model of IUPAC Periodic Table of Elements, also can search an element’s details (period, group, symbol, electron configuration, atomic weight, boiling point and melting point.) through its name, symbol, or atomic number by giving specific inputs. It also displays history of periodic table, some information and tips about the periodic table. All of these will help the user to learn and know the periodic table of IUPAC. It will be interesting to learn periodic table through C programming.

**Features:**

1. Standard view of the IUPAC modern periodic table.

2. Provides historical information about modern periodic table.

3. Provides information about how the modern periodic table is organized.

4. There is an option to search an element of modern periodic table.

5. A search can be done by three ways.

1. By using element’s name.
2. By using element’s symbolic name.
3. By using element’s atomic number.

For that, searching is user friendly.

6. For every successful search the program provides necessary information like name, origin of name, atomic weight, melting point, boiling point, electronic configuration, period number, and group name about that element.

7. All the information are added from reliable sources.

**The program can be controlled by a user driven menu ; that, has these following features :**

1. View of modern periodic table.

2. History of periodic table.

3. Search element of periodic table.

4. Information about periodic table.

5. Exit.

**1. If 1st option is chosen:**

i. Displays the periodic table.

ii. Shows option to go back to the menu & go for the option no 3.

**2. If 2nd option is chosen:**

User finds some notes about history of the periodic table and an option to go back to menu.

**3. If 3rd option is chosen:**

**i.** Displays 3 options to search the element for details.

1. Enter name according to order  takes input  Element’s details

2. Enter symbolic name  takes input  Element’s details

3. Enter atomic number  takes input  Element’s details

**ii.** Shows option to search again.

**iii.** Shows option to go back to menu.

**4. If 4th option is chosen:**

- Displays some tips and information about periodic table and option to go back to menu.

**5. If 5th option is chosen:**

- Ends the program and shows a “thank you” message on console.

**Functions:**

**I**. Pre-Defined Functions :

1. printf( ) : displays information on console

2. scanf( ) : reads the information entered by user from console

3. fopen( ) : opens file

4. fgetc( ) : reads the information (character) from a file

5. strycpy( ): copies one string to another string.

6. system( ): Execute system command

7. fclose( ) : closes file

8. fflush( ): typically used for output stream only

9. exit( ): Terminate calling process

**II.** UserDefined Functions :

1. menu ( ): Is used to displays Menu.
2. displayTable ( ): Is used to display the periodic table.
3. displayInfo ( ): Is used to display the history of periodic table).
4. searchElements( ): Is used takes input to search elements of the table and display the result .
5. Info ( ): Is used to display specific information about periodic table.
6. TAG( ) : Is used to display “Modern Periodic Table”.
7. doMagic( ): function for changing 1st letter of string to capital & rest to small (helps to make searching easy)

**SOURCE CODE**

*#include <stdio.h>*

*#include <stdlib.h>*

*#include <string.h>*

*#include <conio.h>*

*#define max 118*

*struct element*

*{*

*int atomicno;*

*char name[30];*

*char symbol[3];*

*char ec[50];*

*int group;*

*int period;*

*char block;*

*float atomicw;*

*char origin[100];*

*float meltp;*

*float boilp;*

*float density;*

*}elem[max];*

*void menu (void);*

*void displayTable (void);*

*void displayInfo (void);*

*void searchElements(void);*

*void Info (void);*

*void TAG(void);*

*void doMagic(char str[]);*

*int main(void)*

*{*

*menu();*

*printf("\n\n\t\t\t\t THANKS FOR USING\n\n\t\t\tFor any need contact us at \n\t\t\tError.404.saeem.nazmul@gmail.com\n\n\t\t\t\tDEVELOPED BY ERROR404!\n\n\n");*

*return 0;*

*}*

*void TAG (void)*

*{*

*printf("\n");*

*printf("\t\t\t\t===========================\n");*

*printf("\t\t\t\t|| MODERN PERIODIC TABLE ||\n");*

*printf("\t\t\t\t===========================\n");*

*}*

*void menu(void)*

*{*

*int op;*

*do{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*TAG();*

*printf("\t\t\t\t|| MENU ||\n");*

*printf("\t\t\t\t===========================\n\n");*

*printf("\t\t\t===============================================\n");*

*printf("\t\t\t||1. View of modern periodic table . ||\n");*

*printf("\t\t\t===============================================\n");*

*printf("\t\t\t||2. History of periodic table . ||\n");*

*printf("\t\t\t===============================================\n");*

*printf("\t\t\t||3. Search element of periodic table . ||\n");*

*printf("\t\t\t===============================================\n");*

*printf("\t\t\t||4. Information about periodic table . ||\n");*

*printf("\t\t\t===============================================\n");*

*printf("\t\t\t||5. Exit . ||\n");*

*printf("\t\t\t===============================================\n\n");*

*printf("Enter valid option : ");*

*fflush(stdin);*

*scanf("%d",&op);*

*if(op==1)*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*displayTable();*

*}*

*else if(op==2)*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*displayInfo();*

*}*

*else if(op==3)*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*searchElements();*

*}*

*else if (op==4)*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*Info();*

*}*

*else if (op==5)*

*{*

*return;*

*}*

*else*

*{*

*printf("\nNot a valid choice");*

*}*

*}while(op!=5);*

*}*

*void displayTable(void)*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*TAG();*

*int op;*

*printf("\t\t1\t\t-------------------------\n");*

*printf(" ---------------------------------------------------------------------------------------------------------------------\n");*

*printf(" |[group] |[01]| [02]| [03]| [04]| [05]| [06]| [07]| [08]| [09]| [10]| [11]| [12]| [13]| [14]| [15]| [16]| [17]| [18]|\n");*

*printf(" |--------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|\n");*

*printf(" |[period]|----| |-----|\n");*

*printf(" | [1]|[H ]| | [He]|\n");*

*printf(" |--------|----|------ ------------------------------|-----|\n");*

*printf(" | [2]|[Li]| [Be]| | [B ]| [C ]| [N ]| [O ]| [F ]| [Ne]|\n");*

*printf(" |--------|----|-----| |-----|-----|-----|-----|-----|-----|\n");*

*printf(" | [3]|[Na]| [Mg]| | [Al]| [Si]| [P ]| [S ]| [Cl]| [Ar]|\n");*

*printf(" |--------|----|-----|-----------------------------------------------------------|-----|-----|-----|-----|-----|-----|\n");*

*printf(" | [4]|[K ]| [Ca]| [Sc]| [Ti]| [V ]| [Cr]| [Mn]| [Fe]| [Co]| [Ni]| [Cu]| [Zn]| [Ga]| [Ge]| [As]| [Se]| [Br]| [Kr]|\n");*

*printf(" |--------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|\n");*

*printf(" | [5]|[Rb]| [Sr]| [Y ]| [Zr]| [Nb]| [Mo]| [Tc]| [Ru]| [Rh]| [Pd]| [Ag]| [Cd]| [In]| [Sn]| [Sb]| [Te]| [I ]| [Xe]|\n");*

*printf(" |--------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|\n");*

*printf(" | [6]|[Cs]| [Ba]| [La]| [Hf]| [Ta]| [W ]| [Re]| [Os]| [Ir]| [Pt]| [Au]| [Hg]| [Tl]| [Pb]| [Bi]| [Po]| [At]| [Rn]|\n");*

*printf(" |--------|----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|\n");*

*printf(" | [7]|[Fr]| [Ra]| [Ac]| [Rf]| [Db]| [Sg]| [Bh]| [Hs]| [Mt]| [Ds]| [Rg]| [Cn]| [Nh]| [Fl]| [Mc]| [Lv]| [Ts]| [Og]|\n");*

*printf(" ---------------------------------------------------------------------------------------------------------------------\n\n\n");*

*printf ("\n\n\n");*

*printf(" \t----------------------------------------------------------------------------------------------------------\n");*

*printf(" \t|[lanthanide series]|[Ce] |[Pr] |[Nd] |[Pm] |[Sm] |[Eu] |[Gd] |[Tb] |[Dy] |[Ho] |[Er] |[Tm] |[Yb] |[Lu] |\n");*

*printf(" \t|--------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|\n");*

*printf(" \t|[Actinide series] |[Th] |[Pa] |[U ] |[Np] |[Pu] |[Am] |[Cm] |[Bk] |[Cf] |[Es] |[Fm] |[Md] |[No] |[Lr] |\n");*

*printf(" \t----------------------------------------------------------------------------------------------------------\n");*

*do*

*{*

*printf("Enter '1' to search or '0' to go back to menu :");*

*printf("\n\n\n Enter '5' to EXIT:");*

*scanf("%d",&op);*

*if(op==0)*

*{*

*menu();*

*}*

*else if(op==1)*

*{*

*searchElements();return;*

*}*

*else if(op==5)*

*{*

*exit(0);*

*}*

*else*

*printf("Wrong choice!!\t Try again\n");*

*}while(op!=0);*

*}*

*void displayInfo(void)*

*{*

*int op;*

*char ch;*

*FILE \*fp;*

*fp = fopen("periodictable.txt", "r"); // read mode*

*if (fp == NULL)*

*{*

*perror("Error while opening the file.\n");*

*exit(EXIT\_FAILURE);*

*}*

*TAG();*

*while((ch = fgetc(fp)) != EOF)*

*printf("%c", ch);*

*fclose(fp);*

*do*

*{*

*printf("Enter 0 to go back to menu :");*

*scanf("%d",&op);*

*if(op==0)*

*{*

*menu();*

*}*

*else*

*printf("Wrong choice!!\t Try again\n");*

*}while(op!=0);{*

*}*

*}*

*void searchElements(void)*

*{*

*int i,op,choice;*

*char name[30];*

*char sym[3];*

*int atmno;*

*printf("\nYou Can search an element using : \n");*

*printf("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");*

*printf("|1. Search by atomic name |\n");*

*printf("|2. Search by atomic symbol |\n");*

*printf("|3. Search by atomic number |\n");*

*printf("|4. Back to menu |\n");*

*printf("|5. Exit |\n");*

*printf("|\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_|\n\n");*

*CHOICE:*

*printf("Enter valid choice : ");*

*scanf("%d",&choice);*

*for(i=1;i<=max;i++)//\_\_\_\_\_\_\_\_\_\_\_ATOM NUMBER*

*{*

*elem[i].atomicno=i;*

*}*

*//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ATOM NAMES\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ATOMIC SYMBOLS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ELECTRON DEFINITION*

*{*

*strcpy(elem[1].name,"Hydrogen"); strcpy(elem[1].symbol,"H"); strcpy(elem[1].ec,"1s1");*

*strcpy(elem[2].name,"Helium"); strcpy(elem[2].symbol,"He"); strcpy(elem[2].ec,"1s2");*

*strcpy(elem[3].name,"Lithium"); strcpy(elem[3].symbol,"Li"); strcpy(elem[3].ec,"[He] 2s1");*

*strcpy(elem[4].name,"Beryllium"); strcpy(elem[4].symbol,"Be"); strcpy(elem[4].ec,"[He] 2s2 ");*

*strcpy(elem[5].name,"Boron"); strcpy(elem[5].symbol,"B"); strcpy(elem[5].ec,"[He] 2s2 2p1");*

*strcpy(elem[6].name,"Carbon"); strcpy(elem[6].symbol,"C"); strcpy(elem[6].ec,"[He] 2s2 2p2");*

*strcpy(elem[7].name,"Nitrogen"); strcpy(elem[7].symbol,"N"); strcpy(elem[7].ec,"[He] 2s2 2p3");*

*strcpy(elem[8].name,"Oxygen"); strcpy(elem[8].symbol,"O"); strcpy(elem[8].ec,"[He] 2s2 2p4");*

*strcpy(elem[9].name,"Fluorine"); strcpy(elem[9].symbol,"F"); strcpy(elem[9].ec,"[He] 2s2 2p5");*

*strcpy(elem[10].name,"Neon"); strcpy(elem[10].symbol,"Ne"); strcpy(elem[10].ec,"[He] 2s2 2p6");*

*strcpy(elem[11].name,"Sodium"); strcpy(elem[11].symbol,"Na"); strcpy(elem[11].ec,"[Ne] 3s1");*

*strcpy(elem[12].name,"Magnesium"); strcpy(elem[12].symbol,"Mg"); strcpy(elem[12].ec,"[Ne] 3s2");*

*strcpy(elem[13].name,"Aluminium"); strcpy(elem[13].symbol,"Al"); strcpy(elem[13].ec,"[Ne] 3s2 3p1");*

*strcpy(elem[14].name,"Silicon"); strcpy(elem[14].symbol,"Si"); strcpy(elem[14].ec,"[Ne] 3s2 3p2");*

*strcpy(elem[15].name,"Phosphorus"); strcpy(elem[15].symbol,"P"); strcpy(elem[15].ec,"[Ne] 3s2 3p3");*

*strcpy(elem[16].name,"Sulfur"); strcpy(elem[16].symbol,"S"); strcpy(elem[16].ec,"[Ne] 3s2 3p4");*

*strcpy(elem[17].name,"Chlorine"); strcpy(elem[17].symbol,"Cl"); strcpy(elem[17].ec,"[Ne] 3s2 3p5");*

*strcpy(elem[18].name,"Argon"); strcpy(elem[18].symbol,"Ar"); strcpy(elem[18].ec,"[Ne] 3s2 3p6");*

*strcpy(elem[19].name,"Potassium"); strcpy(elem[19].symbol,"K"); strcpy(elem[19].ec,"[Ar] 4s1");*

*strcpy(elem[20].name,"Calcium"); strcpy(elem[20].symbol,"Ca"); strcpy(elem[20].ec,"[Ar] 4s2");*

*strcpy(elem[21].name,"Scandium"); strcpy(elem[21].symbol,"Sc"); strcpy(elem[21].ec,"[Ar] 3d1 4s2");*

*strcpy(elem[22].name,"Titanium"); strcpy(elem[22].symbol,"Ti"); strcpy(elem[22].ec,"[Ar] 3d2 4s2");*

*strcpy(elem[23].name,"Vanadium"); strcpy(elem[23].symbol,"V"); strcpy(elem[23].ec,"[Ar] 3d3 4s2");*

*strcpy(elem[24].name,"Chromium"); strcpy(elem[24].symbol,"Cr"); strcpy(elem[24].ec,"[Ar] 3d5 4s1");*

*strcpy(elem[25].name,"Manganese"); strcpy(elem[25].symbol,"Mn"); strcpy(elem[25].ec,"[Ar] 3d5 4s2");*

*strcpy(elem[26].name,"Iron"); strcpy(elem[26].symbol,"Fe"); strcpy(elem[26].ec,"[Ar] 3d6 4s2");*

*strcpy(elem[27].name,"Cobalt"); strcpy(elem[27].symbol,"Co"); strcpy(elem[27].ec,"[Ar] 3d7 4s2");*

*strcpy(elem[28].name,"Nickel"); strcpy(elem[28].symbol,"Ni"); strcpy(elem[28].ec,"[Ar] 3d8 4s2");*

*strcpy(elem[29].name,"Copper"); strcpy(elem[29].symbol,"Cu"); strcpy(elem[29].ec,"[Ar] 3d10 4s1");*

*strcpy(elem[30].name,"Zinc"); strcpy(elem[30].symbol,"Zn"); strcpy(elem[30].ec,"[Ar] 3d10 4s2");*

*strcpy(elem[31].name,"Gallium"); strcpy(elem[31].symbol,"Ga"); strcpy(elem[31].ec,"[Ar] 3d10 4s2 4p1");*

*strcpy(elem[32].name,"Germanium"); strcpy(elem[32].symbol,"Ge"); strcpy(elem[32].ec,"[Ar] 3d10 4s2 4p2");*

*strcpy(elem[33].name,"Arsenic"); strcpy(elem[33].symbol,"As"); strcpy(elem[33].ec,"[Ar] 3d10 4s2 4p3");*

*strcpy(elem[34].name,"Selenium"); strcpy(elem[34].symbol,"Se"); strcpy(elem[34].ec,"[Ar] 3d10 4s2 4p4");*

*strcpy(elem[35].name,"Bromine"); strcpy(elem[35].symbol,"Br"); strcpy(elem[35].ec,"[Ar] 3d10 4s2 4p5 ");*

*strcpy(elem[36].name,"Krypton"); strcpy(elem[36].symbol,"Kr"); strcpy(elem[36].ec,"[Ar] 3d10 4s2 4p6 ");*

*strcpy(elem[37].name,"Rubidium"); strcpy(elem[37].symbol,"Rb"); strcpy(elem[37].ec,"[Kr] 5s1");*

*strcpy(elem[38].name,"Strontium"); strcpy(elem[38].symbol,"Sr"); strcpy(elem[38].ec,"[Kr] 5s2");*

*strcpy(elem[39].name,"Yttrium"); strcpy(elem[39].symbol,"Y"); strcpy(elem[39].ec,"[Kr] 4d1 5s2");*

*strcpy(elem[40].name,"Zirconium"); strcpy(elem[40].symbol,"Zr"); strcpy(elem[40].ec,"[Kr] 4d2 5s2");*

*strcpy(elem[41].name,"Niobium"); strcpy(elem[41].symbol,"Nb"); strcpy(elem[41].ec,"[Kr] 4d4 5s1");*

*strcpy(elem[42].name,"Molybdenum"); strcpy(elem[42].symbol,"Mo"); strcpy(elem[42].ec,"[Kr] 4d5 5s1");*

*strcpy(elem[43].name,"Technetium"); strcpy(elem[43].symbol,"Tc"); strcpy(elem[43].ec,"[Kr] 4d5 5s2");*

*strcpy(elem[44].name,"Ruthenium"); strcpy(elem[44].symbol,"Ru"); strcpy(elem[44].ec,"[Kr] 4d7 5s1");*

*strcpy(elem[45].name,"Rhodium"); strcpy(elem[45].symbol,"Rh"); strcpy(elem[45].ec,"[Kr] 4d8 5s1");*

*strcpy(elem[46].name,"Palladium"); strcpy(elem[46].symbol,"Pd"); strcpy(elem[46].ec,"[Kr] 4d10");*

*strcpy(elem[47].name,"Silver"); strcpy(elem[47].symbol,"Ag"); strcpy(elem[47].ec,"[Kr] 4d10 5s1");*

*strcpy(elem[48].name,"Cadmium"); strcpy(elem[48].symbol,"Cd"); strcpy(elem[48].ec,"[Kr] 4d10 5s2");*

*strcpy(elem[49].name,"Indium"); strcpy(elem[49].symbol,"In"); strcpy(elem[49].ec,"[Kr] 4d10 5s2 5p1");*

*strcpy(elem[50].name,"Tin"); strcpy(elem[50].symbol,"Sn"); strcpy(elem[50].ec,"[Kr] 4d10 5s2 5p2");*

*strcpy(elem[51].name,"Antimony"); strcpy(elem[51].symbol,"Sb"); strcpy(elem[51].ec,"[Kr] 4d10 5s2 5p3");*

*strcpy(elem[52].name,"Tellurium"); strcpy(elem[52].symbol,"Te"); strcpy(elem[52].ec,"[Kr] 4d10 5s2 5p4");*

*strcpy(elem[53].name,"Iodine"); strcpy(elem[53].symbol,"I"); strcpy(elem[53].ec,"[Kr] 4d10 5s2 5p5");*

*strcpy(elem[54].name,"Xenon"); strcpy(elem[54].symbol,"Xe"); strcpy(elem[54].ec,"[Kr] 4d10 5s2 5p6");*

*strcpy(elem[55].name,"Caesium"); strcpy(elem[55].symbol,"Cs"); strcpy(elem[55].ec,"[Xe] 6s1");*

*strcpy(elem[56].name,"Barium"); strcpy(elem[56].symbol,"Ba"); strcpy(elem[56].ec,"[Xe] 6s2");*

*strcpy(elem[57].name,"Lanthanum"); strcpy(elem[57].symbol,"La"); strcpy(elem[57].ec,"[Xe] 5d1 6s2");*

*strcpy(elem[58].name,"Cerium"); strcpy(elem[58].symbol,"Ce"); strcpy(elem[58].ec,"[Xe] 4f1 5d1 6s2");*

*strcpy(elem[59].name,"Praseodymium"); strcpy(elem[59].symbol,"Pr"); strcpy(elem[59].ec,"[Xe] 4f3 6s2");*

*strcpy(elem[60].name,"Neodymium"); strcpy(elem[60].symbol,"Nd"); strcpy(elem[60].ec,"[Xe] 4f4 6s2");*

*strcpy(elem[61].name,"Promethium"); strcpy(elem[61].symbol,"Pm"); strcpy(elem[61].ec,"[Xe] 4f5 6s2");*

*strcpy(elem[62].name,"Samarium"); strcpy(elem[62].symbol,"Sm"); strcpy(elem[62].ec,"[Xe] 4f6 6s2");*

*strcpy(elem[63].name,"Europium"); strcpy(elem[63].symbol,"Eu"); strcpy(elem[63].ec,"[Xe] 4f7 6s2");*

*strcpy(elem[64].name,"Gadolinium"); strcpy(elem[64].symbol,"Gd"); strcpy(elem[64].ec,"[Xe] 4f7 5d1 6s2");*

*strcpy(elem[65].name,"Terbium"); strcpy(elem[65].symbol,"Tb"); strcpy(elem[65].ec,"[Xe] 4f9 6s2");*

*strcpy(elem[66].name,"Dysprosium"); strcpy(elem[66].symbol,"Dy"); strcpy(elem[66].ec,"[Xe] 4f10 6s2");*

*strcpy(elem[67].name,"Holmium"); strcpy(elem[67].symbol,"Ho"); strcpy(elem[67].ec,"[Xe] 4f11 6s2");*

*strcpy(elem[68].name,"Erbium"); strcpy(elem[68].symbol,"Er"); strcpy(elem[68].ec,"[Xe] 4f12 6s2");*

*strcpy(elem[69].name,"Thulium"); strcpy(elem[69].symbol,"Tm"); strcpy(elem[69].ec,"[Xe] 4f13 6s2");*

*strcpy(elem[70].name,"Ytterbium"); strcpy(elem[70].symbol,"Yb"); strcpy(elem[70].ec,"[Xe] 4f14 6s2");*

*strcpy(elem[71].name,"Lutetium"); strcpy(elem[71].symbol,"Lu"); strcpy(elem[71].ec,"[Xe] 4f14 5d1 6s2");*

*strcpy(elem[72].name,"Hafnium"); strcpy(elem[72].symbol,"Hf"); strcpy(elem[72].ec,"[Xe] 4f14 5d2 6s2");*

*strcpy(elem[73].name,"Tantalum"); strcpy(elem[73].symbol,"Ta"); strcpy(elem[73].ec,"[Xe] 4f14 5d3 6s2");*

*strcpy(elem[74].name,"Tungsten"); strcpy(elem[74].symbol,"W"); strcpy(elem[74].ec,"[Xe] 4f14 5d4 6s2");*

*strcpy(elem[75].name,"Rhenium"); strcpy(elem[75].symbol,"Re"); strcpy(elem[75].ec,"[Xe] 4f14 5d5 6s2");*

*strcpy(elem[76].name,"Osmium"); strcpy(elem[76].symbol,"Os"); strcpy(elem[76].ec,"[Xe] 4f14 5d6 6s2");*

*strcpy(elem[77].name,"Iridium"); strcpy(elem[77].symbol,"Ir"); strcpy(elem[77].ec,"[Xe] 4f14 5d7 6s2");*

*strcpy(elem[78].name,"Platinum"); strcpy(elem[78].symbol,"Pt"); strcpy(elem[78].ec,"[Xe] 4f14 5d9 6s1");*

*strcpy(elem[79].name,"Gold"); strcpy(elem[79].symbol,"Au"); strcpy(elem[79].ec,"[Xe] 4f14 5d10 6s1");*

*strcpy(elem[80].name,"Mercury"); strcpy(elem[80].symbol,"Hg"); strcpy(elem[80].ec,"[Xe] 4f14 5d10 6s2 ");*

*strcpy(elem[81].name,"Thallium"); strcpy(elem[81].symbol,"Tl"); strcpy(elem[81].ec,"[Xe] 4f14 5d10 6s2 6p1");*

*strcpy(elem[82].name,"Lead"); strcpy(elem[82].symbol,"Pb"); strcpy(elem[82].ec,"[Xe] 4f14 5d10 6s2 6p2");*

*strcpy(elem[83].name,"Bismuth"); strcpy(elem[83].symbol,"Bi"); strcpy(elem[83].ec,"[Xe] 4f14 5d10 6s2 6p3");*

*strcpy(elem[84].name,"Polonium"); strcpy(elem[84].symbol,"Po"); strcpy(elem[84].ec,"[Xe] 4f14 5d10 6s2 6p4");*

*strcpy(elem[85].name,"Astatine"); strcpy(elem[85].symbol,"At"); strcpy(elem[85].ec,"[Xe] 4f14 5d10 6s2 6p5");*

*strcpy(elem[86].name,"Radon"); strcpy(elem[86].symbol,"Rn"); strcpy(elem[86].ec,"[Xe] 4f14 5d10 6s2 6p6");*

*strcpy(elem[87].name,"Francium"); strcpy(elem[87].symbol,"Fr"); strcpy(elem[87].ec,"[Rn] 7s1");*

*strcpy(elem[88].name,"Radium"); strcpy(elem[88].symbol,"Ra"); strcpy(elem[88].ec,"[Rn] 7s2");*

*strcpy(elem[89].name,"Actinium"); strcpy(elem[89].symbol,"Ac"); strcpy(elem[89].ec,"[Rn] 6d1 7s2");*

*strcpy(elem[90].name,"Thorium"); strcpy(elem[90].symbol,"Th"); strcpy(elem[90].ec,"[Rn] 6d2 7s2");*

*strcpy(elem[91].name,"Protactinium"); strcpy(elem[91].symbol,"Pa"); strcpy(elem[91].ec,"[Rn] 5f2 6d1 7s2");*

*strcpy(elem[92].name,"Uranium"); strcpy(elem[92].symbol,"U"); strcpy(elem[92].ec,"[Rn] 5f3 6d1 7s2");*

*strcpy(elem[93].name,"Neptunium"); strcpy(elem[93].symbol,"Np"); strcpy(elem[93].ec,"[Rn] 5f4 6d1 7s2");*

*strcpy(elem[94].name,"Plutonium"); strcpy(elem[94].symbol,"Pu"); strcpy(elem[94].ec,"[Rn] 5f6 7s2");*

*strcpy(elem[95].name,"Americium"); strcpy(elem[95].symbol,"Am"); strcpy(elem[95].ec,"[Rn] 5f7 7s2");*

*strcpy(elem[96].name,"Curium"); strcpy(elem[96].symbol,"Cm"); strcpy(elem[96].ec,"[Rn] 5f7 6d1 7s2");*

*strcpy(elem[97].name,"Berkelium"); strcpy(elem[97].symbol,"Bk"); strcpy(elem[97].ec,"[Rn] 5f9 7s2");*

*strcpy(elem[98].name,"Californium"); strcpy(elem[98].symbol,"Cf"); strcpy(elem[98].ec,"[Rn] 5f10 7s2");*

*strcpy(elem[99].name,"Einsteinium"); strcpy(elem[99].symbol,"Es"); strcpy(elem[99].ec,"[Rn] 5f11 7s2");*

*strcpy(elem[100].name,"Fermium"); strcpy(elem[100].symbol,"Fm"); strcpy(elem[100].ec,"[Rn] 5f12 7s2");*

*strcpy(elem[101].name,"Mendelevium"); strcpy(elem[101].symbol,"Md"); strcpy(elem[101].ec,"[Rn] 5f13 7s2");*

*strcpy(elem[102].name,"Nobelium"); strcpy(elem[102].symbol,"No"); strcpy(elem[102].ec,"[Rn] 5f14 7s2");*

*strcpy(elem[103].name,"Lawrencium"); strcpy(elem[103].symbol,"Lr"); strcpy(elem[103].ec,"[Rn] 5f14 7s2 7p1");*

*strcpy(elem[104].name,"Rutherfordium"); strcpy(elem[104].symbol,"Rf"); strcpy(elem[104].ec,"[Rn] 5f14 6d2 7s2");*

*strcpy(elem[105].name,"Dubnium"); strcpy(elem[105].symbol,"Db"); strcpy(elem[105].ec,"[Rn] 5f14 6d3 7s2 ");*

*strcpy(elem[106].name,"Seaborgium"); strcpy(elem[106].symbol,"Sg"); strcpy(elem[106].ec,"[Rn] 5f14 6d4 7s2");*

*strcpy(elem[107].name,"Bohrium"); strcpy(elem[107].symbol,"Bh"); strcpy(elem[107].ec,"[Rn] 5f14 6d5 7s2");*

*strcpy(elem[108].name,"Hassium"); strcpy(elem[108].symbol,"Hs"); strcpy(elem[108].ec,"[Rn] 5f14 6d6 7s2");*

*strcpy(elem[109].name,"Meitnerium"); strcpy(elem[109].symbol,"Mt"); strcpy(elem[109].ec,"[Rn] 5f14 6d7 7s2");*

*strcpy(elem[110].name,"Darmstadtium"); strcpy(elem[110].symbol,"Ds"); strcpy(elem[110].ec,"[Rn] 5f14 6d8 7s2");*

*strcpy(elem[111].name,"Roentgenium"); strcpy(elem[111].symbol,"Rg"); strcpy(elem[111].ec,"[Rn] 5f14 6d9 7s2");*

*strcpy(elem[112].name,"Copernicium"); strcpy(elem[112].symbol,"Cn"); strcpy(elem[112].ec,"[Rn] 5f14 6d10 7s2");*

*strcpy(elem[113].name,"Nihonium"); strcpy(elem[113].symbol,"Nh"); strcpy(elem[113].ec,"[Rn] 5f14 6d10 7s2 7p1");*

*strcpy(elem[114].name,"Flerovium"); strcpy(elem[114].symbol,"Fl"); strcpy(elem[114].ec,"[Rn] 5f14 6d10 7s2 7p2");*

*strcpy(elem[115].name,"Moscovium"); strcpy(elem[115].symbol,"Mc"); strcpy(elem[115].ec,"[Rn] 5f14 6d9 7s2 7p3");*

*strcpy(elem[116].name,"Livermorium"); strcpy(elem[116].symbol,"Lv"); strcpy(elem[116].ec,"[Rn] 5f14 6d10 7s2 7p4");*

*strcpy(elem[117].name,"Tennessine"); strcpy(elem[117].symbol,"Ts"); strcpy(elem[117].ec,"[Rn] 5f14 6d10 7s2 7p5");*

*strcpy(elem[118].name,"Oganesson"); strcpy(elem[118].symbol,"Og"); strcpy(elem[118].ec,"[Rn] 5f14 6d10 7s2 7p6");*

*}*

*//\_\_\_\_\_\_\_\_\_\_\_\_\_\_MELTING POINT\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_BOILING POINT\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_DENSITY\_\_\_\_\_\_\_\_\_\_\_\_\_*

*{*

*elem[1].meltp=14.01; elem[1].boilp=20.28; elem[1].density=0.00008988;*

*elem[2].meltp=0.95; elem[2].boilp=4.22; elem[2].density=0.0001785;*

*elem[3].meltp=453.69; elem[3].boilp=1560; elem[3].density=0.534;*

*elem[4].meltp=1560; elem[4].boilp=2742; elem[4].density=1.85;*

*elem[5].meltp=2349; elem[5].boilp=4200; elem[5].density=2.34;*

*elem[6].meltp=3800; elem[6].boilp=4300; elem[6].density=2.267;*

*elem[7].meltp=63.15; elem[7].boilp=77.36; elem[7].density=0.0012506;*

*elem[8].meltp=54.36; elem[8].boilp=90.20; elem[8].density=0.001429;*

*elem[9].meltp=53.53; elem[9].boilp=85.03; elem[9].density=0.001696;*

*elem[10].meltp=24.56; elem[10].boilp=27.07; elem[10].density=0.0008999;*

*elem[11].meltp=370.87; elem[11].boilp=1156; elem[11].density=0.971;*

*elem[12].meltp=923; elem[12].boilp=1363; elem[12].density=1.738;*

*elem[13].meltp=933.47; elem[13].boilp=2792; elem[13].density=2.698;*

*elem[14].meltp=1687; elem[14].boilp=3538; elem[14].density=2.3296;*

*elem[15].meltp=317.30; elem[15].boilp=550; elem[15].density=1.82;*

*elem[16].meltp=388.36; elem[16].boilp=717.87; elem[16].density=2.067;*

*elem[17].meltp=171.6; elem[17].boilp=239.11; elem[17].density=0.003214;*

*elem[18].meltp=83.80; elem[18].boilp=87.30; elem[18].density=0.0017837;*

*elem[19].meltp=336.53; elem[19].boilp=1032; elem[19].density=0.862;*

*elem[20].meltp=1115; elem[20].boilp=1757; elem[20].density=1.54;*

*elem[21].meltp=1814; elem[21].boilp=3109; elem[21].density=2.989;*

*elem[22].meltp=1941; elem[22].boilp=3560; elem[22].density=4.54;*

*elem[23].meltp=2183; elem[23].boilp=3680; elem[23].density=6.11;*

*elem[24].meltp=2180; elem[24].boilp=2944; elem[24].density=7.15;*

*elem[25].meltp=1519; elem[25].boilp=2334; elem[25].density=7.44;*

*elem[26].meltp=1811; elem[26].boilp=3134; elem[26].density=7.874;*

*elem[27].meltp=1768; elem[27].boilp=3200; elem[27].density=8.86;*

*elem[28].meltp=1728; elem[28].boilp=3186; elem[28].density=8.912;*

*elem[29].meltp=1357.77; elem[29].boilp=2835; elem[29].density=8.96;*

*elem[30].meltp=692.88; elem[30].boilp=1180; elem[30].density=7.134;*

*elem[31].meltp=302.9146; elem[31].boilp=2477; elem[31].density=5.907;*

*elem[32].meltp=1211.40; elem[32].boilp=3106; elem[32].density=5.323;*

*elem[33].meltp=1090; elem[33].boilp=887; elem[33].density=5.776;*

*elem[34].meltp=453; elem[34].boilp=958; elem[34].density=4.809;*

*elem[35].meltp=265.8; elem[35].boilp=332.0; elem[35].density=3.122;*

*elem[36].meltp=115.79; elem[36].boilp=119.93; elem[36].density=0.003733;*

*elem[37].meltp=312.46; elem[37].boilp=961; elem[37].density=1.532;*

*elem[38].meltp=1050; elem[38].boilp=1655; elem[38].density=2.64;*

*elem[39].meltp=1799; elem[39].boilp=3609; elem[39].density=4.469;*

*elem[40].meltp=2128; elem[40].boilp=4682; elem[40].density=6.506;*

*elem[41].meltp=2750; elem[41].boilp=5017; elem[41].density=8.57;*

*elem[42].meltp=2896; elem[42].boilp=4912; elem[42].density=10.22;*

*elem[43].meltp=2430; elem[43].boilp=4538; elem[43].density=11.5;*

*elem[44].meltp=2607; elem[44].boilp=4423; elem[44].density=12.37;*

*elem[45].meltp=2237; elem[45].boilp=3968; elem[45].density=12.41;*

*elem[46].meltp=1828.05; elem[46].boilp=3236; elem[46].density=12.02;*

*elem[47].meltp=1234.93; elem[47].boilp=2435; elem[47].density=10.501;*

*elem[48].meltp=594.22; elem[48].boilp=1040; elem[48].density=8.69;*

*elem[49].meltp=429.75; elem[49].boilp=2345; elem[49].density=7.31;*

*elem[50].meltp=505.08; elem[50].boilp=2875; elem[50].density=7.287;*

*elem[51].meltp=903.78; elem[51].boilp=1860; elem[51].density=6.685;*

*elem[52].meltp=722.66; elem[52].boilp=1261; elem[52].density=6.232;*

*elem[53].meltp=386.85; elem[53].boilp=457.4; elem[53].density=4.93;*

*elem[54].meltp=161.4; elem[54].boilp=165.03; elem[54].density=0.005887;*

*elem[55].meltp=301.59; elem[55].boilp=944; elem[55].density=1.873;*

*elem[56].meltp=1000; elem[56].boilp=2170; elem[56].density=3.594;*

*elem[57].meltp=1193; elem[57].boilp=3737; elem[57].density=6.145;*

*elem[58].meltp=1068; elem[58].boilp=3716; elem[58].density=6.77;*

*elem[59].meltp=1208; elem[59].boilp=3793; elem[59].density=6.773;*

*elem[60].meltp=1297; elem[60].boilp=3347; elem[60].density=7.007;*

*elem[61].meltp=1315; elem[61].boilp=3273; elem[61].density=7.26;*

*elem[62].meltp=1345; elem[62].boilp=2067; elem[62].density=7.52;*

*elem[63].meltp=1099; elem[63].boilp=1802; elem[63].density=5.243;*

*elem[64].meltp=1585; elem[64].boilp=3546; elem[64].density=7.895;*

*elem[65].meltp=1629; elem[65].boilp=3503; elem[65].density=8.229;*

*elem[66].meltp=1680; elem[66].boilp=2840; elem[66].density=8.55;*

*elem[67].meltp=1734; elem[67].boilp=2993; elem[67].density=8.795;*

*elem[68].meltp=1802; elem[68].boilp=3141; elem[68].density=9.066;*

*elem[69].meltp=1818; elem[69].boilp=2223; elem[69].density=9.321;*

*elem[70].meltp=1097; elem[70].boilp=1469; elem[70].density=6.965;*

*elem[71].meltp=1925; elem[71].boilp=3675; elem[71].density=9.84;*

*elem[72].meltp=2506; elem[72].boilp=4876; elem[72].density=13.31;*

*elem[73].meltp=3290; elem[73].boilp=5731; elem[73].density=16.654;*

*elem[74].meltp=3695; elem[74].boilp=5828; elem[74].density=19.25;*

*elem[75].meltp=3459; elem[75].boilp=5869; elem[75].density=21.02;*

*elem[76].meltp=3306; elem[76].boilp=5285; elem[76].density=22.61;*

*elem[77].meltp=2719; elem[77].boilp=4701; elem[77].density=22.56;*

*elem[78].meltp=2041.4; elem[78].boilp=4098; elem[78].density=21.46;*

*elem[79].meltp=1337.33; elem[79].boilp=3129; elem[79].density=19.282;*

*elem[80].meltp=234.43; elem[80].boilp=629.88; elem[80].density=13.5336;*

*elem[81].meltp=577; elem[81].boilp=1746; elem[81].density=11.85;*

*elem[82].meltp=600.61; elem[82].boilp=2022; elem[82].density=11.342;*

*elem[83].meltp=544.7; elem[83].boilp=1837; elem[83].density=9.807;*

*elem[84].meltp=527; elem[84].boilp=1235; elem[84].density=9.32;*

*elem[85].meltp=575; elem[85].boilp=610; elem[85].density=7;*

*elem[86].meltp=202; elem[86].boilp=211.3; elem[86].density=0.00973;*

*elem[87].meltp=300; elem[87].boilp=950; elem[87].density=1.87;*

*elem[88].meltp=973; elem[88].boilp=2010; elem[88].density=5.5;*

*elem[89].meltp=1323; elem[89].boilp=3471; elem[89].density=10.07;*

*elem[90].meltp=2115; elem[90].boilp=5061; elem[90].density=11.72;*

*elem[91].meltp=1841; elem[91].boilp=4300; elem[91].density=15.37;*

*elem[92].meltp=1405.3; elem[92].boilp=4404; elem[92].density=18.95;*

*elem[93].meltp=917; elem[93].boilp=4273; elem[93].density=20.45;*

*elem[94].meltp=912.5; elem[94].boilp=3501; elem[94].density=19.84;*

*elem[95].meltp=1449; elem[95].boilp=2880; elem[95].density=13.69;*

*elem[96].meltp=1613; elem[96].boilp=3383; elem[96].density=13.51;*

*elem[97].meltp=1259; elem[97].boilp=2900; elem[97].density=14.79;*

*elem[98].meltp=1173; elem[98].boilp=1743; elem[98].density=15.1;*

*elem[99].meltp=1133; elem[99].boilp=1269; elem[99].density=8.84;*

*elem[100].meltp=1125; elem[100].boilp=0; elem[100].density=0;*

*elem[101].meltp=1100; elem[101].boilp=0; elem[101].density=0;*

*elem[102].meltp=1100; elem[102].boilp=0; elem[102].density=0;*

*elem[103].meltp=1900; elem[103].boilp=0; elem[103].density=0;*

*elem[104].meltp=2400; elem[104].boilp=5800; elem[104].density=23.2;*

*elem[105].meltp=0; elem[105].boilp=0; elem[105].density=29.3;*

*elem[106].meltp=0; elem[106].boilp=0; elem[106].density=35.0;*

*elem[107].meltp=0; elem[107].boilp=0; elem[107].density=37.1;*

*elem[108].meltp=0; elem[108].boilp=0; elem[108].density=40.7;*

*elem[109].meltp=0; elem[109].boilp=0; elem[109].density=37.4;*

*elem[110].meltp=0; elem[110].boilp=0; elem[110].density=34.8;*

*elem[111].meltp=0; elem[111].boilp=0; elem[111].density=28.7;*

*elem[112].meltp=0; elem[112].boilp=357 ; elem[112].density=23.7;*

*elem[113].meltp=700; elem[113].boilp=1400; elem[113].density=16;*

*elem[114].meltp=340; elem[114].boilp=420; elem[114].density=14;*

*elem[115].meltp=0; elem[115].boilp=0; elem[115].density=0;*

*elem[116].meltp=0; elem[116].boilp=0 ; elem[116].density=0;*

*elem[117].meltp=0; elem[117].boilp=0; elem[117].density=0;*

*elem[118].meltp=0; elem[118].boilp=0; elem[118].density=0;*

*}*

*//Initializing atomic weight, atomic radius and electronegativity.*

*{*

*//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_1 to 40\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_41 to 80\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_81 to 118\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*elem[1].atomicw=1.008; elem[41].atomicw=92.90638; elem[81].atomicw=204.38;*

*elem[2].atomicw=4.002602; elem[42].atomicw=95.96; elem[82].atomicw=207.2;*

*elem[3].atomicw=6.94; elem[43].atomicw=98; elem[83].atomicw=208.98040;*

*elem[4].atomicw=9.012182; elem[44].atomicw=101.07; elem[84].atomicw=209;*

*elem[5].atomicw=10.81; elem[45].atomicw=102.90550; elem[85].atomicw=210;*

*elem[6].atomicw=12.011; elem[46].atomicw=106.42; elem[86].atomicw=222;*

*elem[7].atomicw=14.007; elem[47].atomicw=107.8682; elem[87].atomicw=223;*

*elem[8].atomicw=15.999; elem[48].atomicw=112.411; elem[88].atomicw=226;*

*elem[9].atomicw=18.9984032; elem[49].atomicw=114.818; elem[89].atomicw=227;*

*elem[10].atomicw=20.1797; elem[50].atomicw=118.710; elem[91].atomicw=231.03588;*

*elem[11].atomicw=22.98976928; elem[51].atomicw=121.760; elem[92].atomicw=238.02891;*

*elem[12].atomicw=24.305; elem[52].atomicw=127.60; elem[93].atomicw=237;*

*elem[13].atomicw=26.9815386; elem[53].atomicw=126.90447; elem[94].atomicw=244;*

*elem[14].atomicw=28.085; elem[54].atomicw=131.293; elem[95].atomicw=243;*

*elem[15].atomicw=30.973762; elem[55].atomicw=132.9054519; elem[96].atomicw=247;*

*elem[16].atomicw=32.06; elem[56].atomicw=137.327; elem[97].atomicw=247;*

*elem[17].atomicw=35.45; elem[57].atomicw=138.90547; elem[98].atomicw=251;*

*elem[18].atomicw=39.948; elem[58].atomicw=140.90765; elem[99].atomicw=252;*

*elem[19].atomicw=39.0983; elem[59].atomicw=140.90765; elem[100].atomicw=257;*

*elem[20].atomicw=40.078; elem[60].atomicw=144.242; elem[101].atomicw=258;*

*elem[21].atomicw=44.955912; elem[61].atomicw=145; elem[102].atomicw=259;*

*elem[22].atomicw=47.867; elem[62].atomicw=150.36; elem[103].atomicw=262;*

*elem[23].atomicw=50.9415; elem[63].atomicw=151.964; elem[104].atomicw=267;*

*elem[24].atomicw=51.9961; elem[64].atomicw=157.25; elem[105].atomicw=268;*

*elem[25].atomicw=54.938045; elem[65].atomicw=158.92535; elem[106].atomicw=269;*

*elem[26].atomicw=55.845; elem[66].atomicw=162.500; elem[107].atomicw=270;*

*elem[27].atomicw=58.933195; elem[67].atomicw=164.93032; elem[108].atomicw=269;*

*elem[28].atomicw=58.6934; elem[68].atomicw=167.259; elem[109].atomicw=278;*

*elem[29].atomicw=63.546; elem[69].atomicw=168.93421; elem[110].atomicw=281;*

*elem[30].atomicw=65.38; elem[70].atomicw=173.054; elem[111].atomicw=281;*

*elem[31].atomicw=69.723; elem[71].atomicw=174.9668; elem[112].atomicw=285;*

*elem[32].atomicw=72.630; elem[72].atomicw=178.49; elem[113].atomicw=286;*

*elem[33].atomicw=74.92160; elem[73].atomicw=180.94788; elem[114].atomicw=289;*

*elem[34].atomicw=78.96; elem[74].atomicw=183.84; elem[115].atomicw=290;*

*elem[35].atomicw=79.904; elem[75].atomicw=186.207; elem[116].atomicw=293;*

*elem[36].atomicw=83.798; elem[76].atomicw=190.23; elem[117].atomicw=294;*

*elem[37].atomicw=85.4678; elem[77].atomicw=192.217; elem[118].atomicw=294;*

*elem[38].atomicw=87.62; elem[78].atomicw=195.084;*

*elem[39].atomicw=88.90585; elem[79].atomicw=196.966569;*

*elem[40].atomicw=91.224; elem[80].atomicw=200.592;*

*}*

*{*

*// ORIGIN NAME \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*strcpy(elem[1].origin,"the Greek 'hydro' and 'genes' meaning water-forming");*

*strcpy(elem[2].origin,"the Greek 'helios' meaning sun");*

*strcpy(elem[3].origin,"the Greek 'lithos' meaning stone");*

*strcpy(elem[4].origin,"the Greek name for beryl, 'beryllo'");*

*strcpy(elem[5].origin,"the Arabic 'buraq', which was the name for borax");*

*strcpy(elem[6].origin,"the Latin 'carbo', meaning charcoal");*

*strcpy(elem[7].origin,"the Greek 'nitron' and 'genes' meaning nitre-forming");*

*strcpy(elem[8].origin,"the Greek 'oxy' and 'genes' meaning acid-forming");*

*strcpy(elem[9].origin,"the Latin 'fluere', meaning to flow");*

*strcpy(elem[10].origin,"the Greek 'neos', meaning new");*

*strcpy(elem[11].origin,"the English word soda");*

*strcpy(elem[12].origin,"Magnesia, a district of Eastern Thessaly in Greece");*

*strcpy(elem[13].origin,"Magnesia, a district of Eastern Thessaly in Greece");*

*strcpy(elem[14].origin,"the Latin 'silex' or 'silicis', meaning flint");*

*strcpy(elem[15].origin,"the Greek 'phosphoros', meaning bringer of light");*

*strcpy(elem[16].origin,"Either from the Sanskrit 'sulvere', or the Latin 'sulfurium', both names for sulfur");*

*strcpy(elem[17].origin,"the Greek 'chloros', meaning greenish yellow");*

*strcpy(elem[18].origin,"the Greek, 'argos', meaning idle");*

*strcpy(elem[19].origin,"the English word potash");*

*strcpy(elem[20].origin,"the Latin 'calx' meaning lime");*

*strcpy(elem[21].origin,"Scandinavia");*

*strcpy(elem[22].origin,"Titans, the sons of the Earth goddess of Greek mythology");*

*strcpy(elem[23].origin,"Vanadis, an old Norse name for the Scandinavian goddess Freyja");*

*strcpy(elem[24].origin,"the Greek 'chroma', meaning colour");*

*strcpy(elem[25].origin,"from the black magnesium oxide, 'magnesia nigra'");*

*strcpy(elem[26].origin,"the Anglo-Saxon name iren");*

*strcpy(elem[27].origin,"the German word 'kobald', meaning goblin");*

*strcpy(elem[28].origin,"the shortened of the German 'kupfernickel' meaning St. Nicholas's copper");*

*strcpy(elem[29].origin,"derived from the Latin 'Cyprium aes', meaning a metal from Cyprus");*

*strcpy(elem[30].origin,"the German, 'zinc', meaning stone");*

*strcpy(elem[31].origin,"France");*

*strcpy(elem[32].origin,"Germany");*

*strcpy(elem[33].origin,"the Greek name 'arsenikon' for the yellow pigment orpiment");*

*strcpy(elem[34].origin,"Moon");*

*strcpy(elem[35].origin,"the Greek 'bromos' meaning stench");*

*strcpy(elem[36].origin,"the Greek 'kryptos', meaning hidden");*

*strcpy(elem[37].origin,"the Latin 'rubidius', meaning deepest red");*

*strcpy(elem[38].origin,"Strontian, a small town in Scotland");*

*strcpy(elem[39].origin,"Ytterby, Sweden");*

*strcpy(elem[40].origin,"the Persian 'zargun', meaning gold coloured");*

*strcpy(elem[41].origin,"Niobe, daughter of king Tantalus from Greek mythology");*

*strcpy(elem[42].origin,"the Greek 'molybdos' meaning lead");*

*strcpy(elem[43].origin,"the Greek 'tekhnetos' meaning artificial");*

*strcpy(elem[44].origin,"Russia");*

*strcpy(elem[45].origin,"the Greek 'rhodon', meaning rose coloured");*

*strcpy(elem[46].origin,"the then recently-discovered asteroid Pallas, considered a planet at the time");*

*strcpy(elem[47].origin,"the Anglo-Saxon name siolfur");*

*strcpy(elem[48].origin,"the Latin name for the mineral calmine, 'cadmia'");*

*strcpy(elem[49].origin,"the Latin 'indicium', meaning violet or indigo");*

*strcpy(elem[50].origin,"the Anglo-Saxon word tin");*

*strcpy(elem[51].origin,"the Greek 'anti – monos', meaning not alone");*

*strcpy(elem[52].origin,"Earth, the third planet on solar system");*

*strcpy(elem[53].origin,"the Greek 'iodes' meaning violet");*

*strcpy(elem[54].origin,"the Greek 'xenos' meaning stranger");*

*strcpy(elem[55].origin,"the Latin 'caesius', meaning sky blue");*

*strcpy(elem[56].origin,"the Greek 'barys', meaning heavy");*

*strcpy(elem[57].origin,"the Greek 'lanthanein', meaning to lie hidden");*

*strcpy(elem[58].origin,"Ceres, the Roman God of agriculture");*

*strcpy(elem[59].origin,"the Greek 'prasios didymos' meaning green twin");*

*strcpy(elem[60].origin,"the Greek 'neos didymos' meaning new twin");*

*strcpy(elem[61].origin,"Prometheus of Greek mythology who stole fire from the Gods and gave it to humans");*

*strcpy(elem[62].origin,"Samarskite, the name of the mineral from which it was first isolated");*

*strcpy(elem[63].origin,"Europe");*

*strcpy(elem[64].origin,"Johan Gadolin, chemist, physicist and mineralogist");*

*strcpy(elem[65].origin,"Ytterby, Sweden");*

*strcpy(elem[66].origin,"the Greek 'dysprositos', meaning hard to get");*

*strcpy(elem[67].origin,"Stockholm, Sweden");*

*strcpy(elem[68].origin,"Ytterby, Sweden");*

*strcpy(elem[69].origin,"Thule, the ancient name for Scandinavia");*

*strcpy(elem[70].origin,"Ytterby, Sweden");*

*strcpy(elem[71].origin,"Paris, France");*

*strcpy(elem[72].origin,"Copenhagen, Denmark");*

*strcpy(elem[73].origin,"King Tantalus, father of Niobe from Greek mythology");*

*strcpy(elem[74].origin,"the Swedish 'tung sten' meaning heavy stone");*

*strcpy(elem[75].origin,"Rhine, a river");*

*strcpy(elem[76].origin,"the Greek 'osme', meaning smell");*

*strcpy(elem[77].origin,"Iris, the Greek goddess of the rainbow");*

*strcpy(elem[78].origin,"the Spanish 'platina', meaning little silver");*

*strcpy(elem[79].origin,"the Anglo-Saxon word gold");*

*strcpy(elem[80].origin,"Mercury, the first planet in the Solar System");*

*strcpy(elem[81].origin,"the Greek 'thallos', meaning a green twig");*

*strcpy(elem[82].origin,"the Anglo-Saxon lead");*

*strcpy(elem[83].origin,"the German 'Bisemutum' a corruption of 'Weisse Masse' meaning white mass");*

*strcpy(elem[84].origin,"Poland, the native country of Marie Curie, who first isolated the element");*

*strcpy(elem[85].origin,"the Greek 'astatos', meaning unstable");*

*strcpy(elem[86].origin,"From radium, as it was first detected as an emission from radium during radioactive decay");*

*strcpy(elem[87].origin,"France, where it was first discovered");*

*strcpy(elem[88].origin,"the Latin 'radius', meaning ray");*

*strcpy(elem[89].origin,"the Greek 'actinos', meaning a ray");*

*strcpy(elem[90].origin,"Thor, the Scandinavian god of thunder");*

*strcpy(elem[91].origin,"the Greek 'protos', as a prefix to the element actinium");*

*strcpy(elem[92].origin,"Uranus, the seventh planet in the Solar System");*

*strcpy(elem[93].origin,"Neptune, the eighth planet in the Solar System");*

*strcpy(elem[94].origin,"Pluto, a dwarf planet in the Solar System");*

*strcpy(elem[95].origin,"Americas, the continent where the element was first synthesized");*

*strcpy(elem[96].origin,"Pierre Curie, a physicist, and Marie Curie, a physicist and chemist");*

*strcpy(elem[97].origin,"Berkeley, California, USA, where the element was first synthesized");*

*strcpy(elem[98].origin,"State of California, USA, where the element was first synthesized");*

*strcpy(elem[99].origin,"Albert Einstein, physicist");*

*strcpy(elem[100].origin,"Enrico Fermi, physicist");*

*strcpy(elem[101].origin,"Dmitri Mendeleyev, chemist and inventor");*

*strcpy(elem[102].origin,"Alfred Nobel, chemist, engineer, innovator, and armaments manufacturer");*

*strcpy(elem[103].origin,"Ernest O. Lawrence, physicist");*

*strcpy(elem[104].origin,"Ernest Rutherford, chemist and physicist");*

*strcpy(elem[105].origin,"Dubna, Russia");*

*strcpy(elem[106].origin,"Glenn T. Seaborg, scientist");*

*strcpy(elem[107].origin,"Hesse, Germany, where the element was first synthesized");*

*strcpy(elem[108].origin,"Lise Meitner, physicist");*

*strcpy(elem[109].origin,"Darmstadt, Germany, where the element was first synthesized");*

*strcpy(elem[110].origin,"Wilhelm Conrad Röntgen, physicist");*

*strcpy(elem[111].origin,"Nicolaus Copernicus, astronomer");*

*strcpy(elem[112].origin,"IUPAC systematic element name");*

*strcpy(elem[113].origin,"Georgy Flyorov, physicist");*

*strcpy(elem[114].origin,"IUPAC systematic element name");*

*strcpy(elem[115].origin,"IUPAC systematic element name");*

*strcpy(elem[116].origin,"IUPAC systematic element name");*

*strcpy(elem[117].origin,"IUPAC systematic element name");*

*strcpy(elem[118].origin,"IUPAC systematic element name");*

*}*

*//\_\_\_\_\_\_\_\_\_\_\_\_GROUP & BLOCK \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*{*

*for(i=1;i<=118;i++)*

*{*

*if(i==1 || i==3 || i==11 || i==19 || i==37 || i==55 || i==87)*

*{*

*elem[i].group=1;*

*elem[i].block='s';*

*}*

*else if(i==4 || i==12 || i==20 || i==38 || i==56 || i==88)*

*{*

*elem[i].group=2;*

*elem[i].block='s';*

*}*

*else if(i==21 || i==39 || i==57 || i==89)*

*{*

*elem[i].group=3;*

*elem[i].block='d';*

*}*

*else if(i==22 || i==40 || i==72 || i==104)*

*{*

*elem[i].group=4;*

*elem[i].block='d';*

*}*

*else if(i==23 || i==41 || i==73 || i==105)*

*{*

*elem[i].group=5;*

*elem[i].block='d';*

*}*

*else if(i==24 || i==42 || i==74 || i==106)*

*{*

*elem[i].group=6;*

*elem[i].block='d';*

*}*

*else if(i==25 || i==43 || i==75 || i==107)*

*{*

*elem[i].group=7;*

*elem[i].block='d';*

*}*

*else if(i==26 || i==44 || i==76 || i==108)*

*{*

*elem[i].group=8;*

*elem[i].block='d';*

*}*

*else if(i==27 || i==45 || i==77 || i==109)*

*{*

*elem[i].group=9;*

*elem[i].block='d';*

*}*

*else if(i==28 || i==46 || i==78 || i==110)*

*{*

*elem[i].group=10;*

*elem[i].block='d';*

*}*

*else if(i==29 || i==47 || i==79 || i==111)*

*{*

*elem[i].group=11;*

*elem[i].block='d';*

*}*

*else if(i==30 || i==48 || i==80 || i==112)*

*{*

*elem[i].group=12;*

*elem[i].block='d';*

*}*

*else if(i==5 || i==13 || i==31 || i==49 || i==81 || i==113)*

*{*

*elem[i].group=13;*

*elem[i].block='p';*

*}*

*else if(i==6 || i==14 || i==32 || i==50 || i==82 || i==114)*

*{*

*elem[i].group=14;*

*elem[i].block='p';*

*}*

*else if(i==7 || i==15 || i==33 || i==51 || i==83 || i==115)*

*{*

*elem[i].group=15;*

*elem[i].block='p';*

*}*

*else if(i==8 || i==16 || i==34 || i==52 || i==84 || i==116)*

*{*

*elem[i].group=16;*

*elem[i].block='p';*

*}*

*else if(i==9 || i==17 || i==35 || i==53 || i==85 || i==117)*

*{*

*elem[i].group=17;*

*elem[i].block='p';*

*}*

*else if(i==2 || i==10 || i==18 || i==36 || i==54 || i==86 || i==118)*

*{*

*elem[i].group=18;*

*elem[i].block='p';*

*}*

*}//FOR LANTHINIDE SERIES AND ACTINIDE SERIES\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*for(i=57;i<=71;++i)*

*{*

*elem[i].block='f';*

*}*

*for(i=89;i<=103;++i)*

*{*

*elem[i].block='f';*

*}*

*//PERIOD NUMBER \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*{*

*for(i=1;i<=2;++i)*

*{*

*elem[i].period=1;*

*}*

*for(i=3;i<=10;++i)*

*{*

*elem[i].period=2;*

*}*

*for(i=11;i<=18;++i)*

*{*

*elem[i].period=3;*

*}*

*for(i=19;i<=36;++i)*

*{*

*elem[i].period=4;*

*}*

*for(i=37;i<=54;++i)*

*{*

*elem[i].period=5;*

*}*

*for(i=55;i<=86;++i)*

*{*

*elem[i].period=6;*

*}*

*for(i=87;i<=118;++i)*

*{*

*elem[i].period=7;*

*}*

*}*

*} // CHOOSING AND SHOWING RESULT\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*if(choice==1) //SEARCH BY NAME \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*{*

*do*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*printf("\nEnter name like [Hydrogen] : ");*

*fflush(stdin);*

*gets(name);*

*doMagic(name);*

*for(i=1;i<=118;i++)*

*{*

*if(strcmp(name,elem[i].name)==0)*

*{*

*printf("\n");*

*printf("\t Name : ");*

*puts(elem[i].name);*

*printf("\n");*

*printf("\t Origin of Name : ");*

*puts(elem[i].origin);*

*printf("\n");*

*printf("\t Atomic Number : %d\n",elem[i].atomicno);*

*printf("\t Group : %d\n",elem[i].group);*

*printf("\t Period : %d\n",elem[i].period);*

*printf("\t Block : %c\n",elem[i].block);*

*printf("\t Atomic Weight : %.2f\n",elem[i].atomicw);*

*printf("\t Melting Point : %.2f K\n",elem[i].meltp);*

*printf("\t Boiling Point : %.2f K\n",elem[i].boilp);*

*printf("\t Density : %.2f g/cm3\n",elem[i].density);*

*printf("\t Electron Definition : ");*

*puts(elem[i].ec);*

*printf("\n");*

*printf("\t If Melting points, Boiling points, Density is 0\n\t It means these behaviours are unknown till now.\n\n\n");*

*break;}*

*}*

*if(i<0||i>118)*

*{*

*printf("\n\n\t\tSORRY! Could not found!\n\n\t\tPlease,check your spelling.\n\n\n");*

*}*

*printf("Enter 0 to go back to menu or press any key to search again : ");*

*fflush(stdin);*

*scanf("%d",&op);*

*if(op==0)*

*{*

*menu();*

*}*

*else*

*{*

*continue ;*

*}*

*}while(op!=0);*

*}*

*else if(choice==2) //SEARCH BY SYMBOL\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*{*

*do*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*fflush(stdin);*

*printf("\nEnter name like [H] : ");*

*gets(sym);*

*doMagic(sym);*

*for(i=1;i<=118;i++)*

*{*

*if(strcmp(sym,elem[i].symbol)==0)*

*{*

*printf("\n");*

*printf("\t Name : ");*

*puts(elem[i].name);*

*printf("\n");*

*printf("\t Origin of Name : ");*

*puts(elem[i].origin);*

*printf("\n");*

*printf("\t Atomic Number : %d\n",elem[i].atomicno);*

*printf("\t Group : %d\n",elem[i].group);*

*printf("\t Period : %d\n",elem[i].period);*

*printf("\t Block : %c\n",elem[i].block);*

*printf("\t Atomic Weight : %.2f\n",elem[i].atomicw);*

*printf("\t Melting Point : %.2f K\n",elem[i].meltp);*

*printf("\t Boiling Point : %.2f K\n",elem[i].boilp);*

*printf("\t Density : %.2f g/cm3\n",elem[i].density);*

*printf("\t Electron Definition : ");*

*puts(elem[i].ec);*

*printf("\n");*

*printf("\t If Melting points, Boiling points, Density is 0\n\t It means these behaviours are unknown till now. \n\n\n");break;*

*}*

*}*

*if(i>118)*

*{printf("\n\n\t\tSORRY! Could not found!\n\n\t\tPlease,check your spelling.\n\n\n");}*

*printf("Enter 0 to go back to menu or press any key to search again : ");*

*scanf("%d",&op);*

*if(op==0)*

*{*

*menu();*

*}*

*else*

*{*

*continue ;*

*}*

*}while(op!=0);*

*}*

*else if(choice==3) //SEARCH BY ATOMIC NUMBER\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*{*

*do*

*{*

*system("cls"); //CLEARING SCREEN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

*printf("\nEnter atomic number : ");*

*scanf("%d",&atmno);*

*for(i=1;i<=118;i++)*

*{*

*if(elem[i].atomicno==atmno)*

*{*

*printf("\n");*

*printf("\t Name : ");*

*puts(elem[i].name);*

*printf("\n");*

*printf("\t Origin of Name : ");*

*puts(elem[i].origin);*

*printf("\n");*

*printf("\t Atomic Number : %d\n",elem[i].atomicno);*

*printf("\t Group : %d\n",elem[i].group);*

*printf("\t Period : %d\n",elem[i].period);*

*printf("\t Block : %c\n",elem[i].block);*

*printf("\t Atomic Weight : %.2f\n",elem[i].atomicw);*

*printf("\t Melting Point : %.2f K\n",elem[i].meltp);*

*printf("\t Boiling Point : %.2f K\n",elem[i].boilp);*

*printf("\t Density : %.2f g/cm3\n",elem[i].density);*

*printf("\t Electron Definition : ");*

*puts(elem[i].ec);*

*printf("\n");*

*printf("\t If Melting points, Boiling points, Density is 0\n\t It means these behaviours are unknown till now.\n\n\n");*

*break;*

*}*

*}*

*printf("Enter 0 to go back to menu or press any key to search again : ");*

*scanf("%d",&op);*

*if(op==0)*

*{*

*menu();*

*}*

*else*

*{*

*continue ;*

*}*

*}while(op!=0);*

*}*

*else if(op==4)*

*{*

*menu();*

*}*

*else if(choice==5)*

*{*

*exit(0);*

*}*

*else*

*{ printf("\n\t\t\t Invalid choice . Please make a valid choice.\n\n");*

*fflush(stdin);*

*goto CHOICE;*

*}*

*}*

*void doMagic(char str[])*

*{int i=0;*

*if(str[0]>='a' && str[0]<='z')*

*str[i]=str[i]-('a'-'A');*

*for(i=1;str[i]!='\0';i++)*

*{*

*if(str[i]>='A' && str[i]<='Z')*

*str[i]=str[i]+('a'-'A');*

*}*

*}*

*void Info (void)*

*{*

*int op;*

*char ch;*

*FILE \*fp;*

*fp = fopen("info.txt", "r"); // READ*

*if (fp == NULL)*

*{*

*perror("Error while opening the file.\n");*

*exit(EXIT\_FAILURE);*

*}*

*TAG();*

*while((ch = fgetc(fp)) != EOF)*

*{fflush(stdin);*

*printf("%c", ch);}*

*fclose(fp);*

*do*

*{*

*printf("Enter 0 to go back to menu :");*

*scanf("%d",&op);*

*if(op==0)*

*{*

*menu();*

*}*

*else*

*printf("Wrong choice!!\t Try again\n");*

*}while(op!=0);*

*}*

//THE\_End\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_